

Assignment-3

1. (a) **False.**

Because \mathbf{A} is nonsingular, there are a number of properties that are associated with it. One of these is the fact that \mathbf{A} being singular implies that $\mathbf{Ax} = \mathbf{b}$ has a unique solution for all \mathbf{b} . Likewise if $\mathbf{b} = \mathbf{0}$, then \mathbf{x} is necessarily the 0 vector.

This question was done assuming that $\mathbf{A} = n \times n \rightarrow \mathbf{b} = 1 \times n$. If \mathbf{b} is of size $m < n$, then $\mathbf{Ax} = \mathbf{b}$ has an infinity of solutions. However, this does not depend on the contents of \mathbf{b} , but rather the initial parameters of the problem.

(b) **False.**

This can be proven false simply by taking the determinant of a 3×3 matrix.

$$\det(\mathbf{A}) = a_{11}(a_{22}a_{33} + a_{23}a_{32}) - a_{12}(a_{21}a_{33} + a_{23}a_{31}) + a_{13}(a_{21}a_{32} + a_{22}a_{31})$$

As there are terms in this equation that are not dependant on a_{ii} , a matrix with a zero on the principle diagonal is not necessarily singular.

(c) **False.**

The conditioning of a matrix \mathbf{A} is related to the largest sum of a single column (the 1-norm). The partial pivoting algorithm scans the rest of the column below the current pivot point (a_{kk}) for the largest value.

A well-conditioned matrix would simply have small column sums, which does not necessarily affect the location of the largest values in each row, which in turn does not affect the need for partial pivoting.

(d) **True.**

By definition, the condition number of a matrix $\mathbf{A} = \|\mathbf{A}\| \|\mathbf{A}^{-1}\|$. Therefore, the condition number of $\mathbf{A}^{-1} = \|\mathbf{A}^{-1}\| \|(\mathbf{A}^{-1})^{-1}\| = \|\mathbf{A}^{-1}\| \|\mathbf{A}\|$. As $\|\mathbf{A}\|$, $\|\mathbf{A}^{-1}\|$ are scalar values, the two products are the same.

(e) **False.**

When working with a linear system, the solution to the system is the point at which every line defined by matrix \mathbf{A} intersects. If there is no point where every line intersects, then there are no solutions. Because the system is linear, two lines can have only one discrete intersection. Therefore, it is not possible for 2 (or more) lines to intersect at more than one point, so there must be either exactly 1 solution, or infinitely many solutions.

(f) One of the reasons pivoting is essential in the G.E. method of solving matrices is to avoid potential zeroes on the principle diagonal. When the G.E. is being done, each element on the principle diagonal determines the multiplier for the rows below. $m_{ij} = \frac{a_{ij}}{a_{jj}}$. Since

the diagonal element is in the denominator, it must be nonzero. This is a mathematical reason for pivoting.

The other reason is a numerical one. If the current pivot point is a_{jj} , and this value is very small (10^{-16}), then the multiplier that results will be extremely large. If the values in column j are then relatively small to a_{jj} , the subtraction will result in a loss of significance, due to not being able to express numbers such as $10^{16} - 1$ accurately. Partial pivoting moves the row with the largest j^{th} element to the current row, then computes the multiplier. This avoids the large multiplier result, and will aid in maintaining significance.

- (g) The condition number is a nontrivial computation due to the computation of the norms that make up the number. In order to compute the 1-norm, which is common, the algorithm must sum every column, the largest of which becomes the condition number. This alone requires $O(n^2)$ operations. The algorithm must then compute the inverse of \mathbf{A} , which is an even more expensive operation. More curcially, the inverse of matrix \mathbf{A} must be completely recomputed if \mathbf{A} is even slightly modified.
- (h) On its own, the relative residual does not carry much weight. A small residual would, at first glance, seem to imply low error, and therefore higher accuracy. However, this is not the case, and it turns out that the condition number of a matrix is directly involved in the accuracy of the solution. In order for a solution to be accurate, the solution should not only have a small relative residual, but should also be moderately well conditioned.
- (i)
 - i. $\det(c\mathbf{A}) = c^n * \det(\mathbf{A})$
 In this case, \mathbf{A} would be an $n \times n$ diagonal matrix, with every diagonal entry = 1, and $c = \frac{1}{2}$. The determinant of a diagonal matrix with every diagonal value = 1 is simply the identity matrix, and $\det(\mathbf{I})$ is 1. Therefore, the determinant of the matrix $\frac{1}{2}\mathbf{I} = \frac{1}{2}^n$.
 - ii. Assume the use of the 1-norm
 $\|\mathbf{A}\|_1 = \frac{1}{2}$
 Because the inverse of a diagonal matrix is found by simply inverting all diagonal elements, $\|\mathbf{A}^{-1}\|_1 = 2$
 $K(\mathbf{A}) = \|\mathbf{A}\|_1 * \|\mathbf{A}^{-1}\|_1 = 1$
 - iii. Any scalar multiple of the identity matrix will not amplify input errors. This makes intuitive sense, as the condition number of a matrix sets an upper bound on the maximum distortion of the result due to input errors. Because each element in the system (x_1, x_2, \dots, x_n) only appears once in \mathbf{A} , each entry simply takes the corresponding \mathbf{b} value ($a_1 = b_1, a_2 = b_2, \dots, a_n = b_n$). Any error that is present in the input will be directly translate to the output.

2. (a)

$$\begin{bmatrix} 1 & 2 \\ 2 & 4.01 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6.01 \end{bmatrix}$$

$$x_1 = 3 - 2x_2$$

$$2(3 - 2x_2) + 4.01x_2 = 6.01$$

$$6 + 0.01x_2 = 6.01$$

$$0.01x_2 = 0.01$$

$$x_2 = 1$$

$$x_1 = 1$$

(b)

$$\begin{bmatrix} 1 & 2 \\ 2 & 4.01 \end{bmatrix} \begin{bmatrix} -600 \\ 301 \end{bmatrix} = \begin{bmatrix} 2 \\ 7.1 \end{bmatrix}$$

Here,

$$\mathbf{x}^* = \begin{bmatrix} -600 \\ 301 \end{bmatrix}$$

and

$$\mathbf{b}^* = \begin{bmatrix} 2 \\ 7.1 \end{bmatrix}$$

The relative forward error is defined by $\frac{\|\mathbf{x}^*\|_\infty}{\|\mathbf{x}\|_\infty}$, while the relative backward error is $\frac{\|\mathbf{b}^*\|_\infty}{\|\mathbf{b}\|_\infty}$.

Relative Forward Error	Relative Backward Error
$\ \mathbf{x}^*\ = 600$	$\ \mathbf{b}^*\ = 7.1$
$\ \mathbf{x}\ = 1$	$\ \mathbf{b}\ = 6.01$
$\frac{600}{1} = 600$	$\frac{7.1}{6.01} \approx 1.18 \dots$

The error magnification factor is (relative forwards error \div relative backwards error).

$$E_m = 600 \div \frac{7.1}{6.01} \approx 507.89 \dots$$

(c) $K(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\|$

$$\|\mathbf{A}\| = 6.01$$

$$\mathbf{A}^{-1} = \frac{1}{0.01} \begin{bmatrix} 4.01 & -2 \\ -2 & 1 \end{bmatrix}$$

$$\|\mathbf{A}^{-1}\| = 201$$

$$K(\mathbf{A}) = 201 * 6.01 = 1208.01$$

The poor conditioning of the number is a clear indication that a small change to the data will cause a large change in the result. In this case, the altered data that gave rise to the vector \mathbf{x}^* was \mathbf{b}^* . The large condition number sets the upper bound on the relative output error, meaning that \mathbf{x}^* can grow extremely large, without producing extremely different results in \mathbf{b}^* .

3. (a) Begin with the augmented matrix $\mathbf{A}|\mathbf{b}$

$$\mathbf{A}|\mathbf{b} = \left[\begin{array}{ccccc} 1 & 2 & 3 & 4 & -6 \\ 2 & 3 & 7 & 9 & -17 \\ -2 & -6 & -2 & -4 & -4 \\ 1 & 0 & 11 & 15 & -34 \end{array} \right]$$

Step 1: Pivot point a_{11}

First, we must employ partial pivoting to ensure there are no multipliers > 1 . We see that row 2 has the largest value in column 1, so we use this as the first row instead. The permutation matrix associated with swapping rows $R1$ and $R2$ is

$$\mathbf{P} = \left[\begin{array}{cccc} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right]$$

and the matrix we are now operating on is

$$\mathbf{A} \left[\begin{array}{ccccc} 2 & 3 & 7 & 9 & -17 \\ 1 & 2 & 3 & 4 & -6 \\ -2 & -6 & -2 & -4 & -4 \\ 1 & 0 & 11 & 15 & -34 \end{array} \right]$$

We now calculate the multipliers needed to zero out the first column.

$$m_{21} = \frac{a_{21}}{a_{11}} = 0.5 \quad \parallel \quad m_{31} = \frac{a_{31}}{a_{11}} = -1 \quad \parallel \quad m_{41} = \frac{a_{41}}{a_{11}} = 0.5$$

Then, carry out the row operations on $R2, R3, R4$.

$$R2 \rightarrow R2 - m_{21}R1 \quad \parallel \quad R3 \rightarrow R3 - m_{31}R1 \quad \parallel \quad R4 \rightarrow R4 - m_{41}R1$$

The new matrix generated after these operations is

$$\mathbf{A} = \left[\begin{array}{ccccc} 2 & 3 & 7 & 9 & -17 \\ (0.5) & 0.5 & -0.5 & -0.5 & -2.5 \\ (-1) & -3 & -5 & 5 & -21 \\ (0.5) & -1.5 & 7.5 & 10.5 & -25.5 \end{array} \right]$$

where the normal zeros are replaced by the multipliers.

Step 2: Pivot point a_{22}

Again, start by swapping rows according to partial pivoting. This includes changing the positions of the multipliers, although their values will not be affected by the row operations that follow. It also includes swapping the corresponding rows in the permutation matrix. In this case, we swap rows $R2$ and $R3$

$$\mathbf{A} = \left[\begin{array}{ccccc} 2 & 3 & 7 & 9 & -17 \\ (-1) & -3 & -5 & 5 & -21 \\ (0.5) & 0.5 & -0.5 & -0.5 & -2.5 \\ (0.5) & -1.5 & 7.5 & 10.5 & -25.5 \end{array} \right]$$

and

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Next, calculate the new multipliers. In this step there are only two, as the only rows that need have items zeroed out are $R3$ and $R4$

$$m_{32} = \frac{a_{32}}{a_{22}} = -\frac{1}{6} \parallel m_{42} = \frac{a_{42}}{a_{22}} = 0.5$$

Next, perform the row operations on $R3, R4$

$$R3 \rightarrow R3 - m_{32}R2 \parallel R4 \rightarrow R4 - m_{42}R2$$

The new matrix is

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & 7 & 9 & -17 \\ (-1) & -3 & -5 & 5 & -21 \\ (0.5) & (-\frac{1}{6}) & \frac{1}{3} & \frac{1}{3} & -6 \\ (0.5) & (0.5) & 5 & 8 & 36 \end{bmatrix}$$

again, with the normal zeros replaced by the multipliers.

Step 3: Pivot point a_{33}

In this case, we have to switch $R3$ and $R4$ in \mathbf{A} and \mathbf{P} according to partial pivoting.

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & 7 & 9 & -17 \\ (-1) & -3 & -5 & 5 & -21 \\ (0.5) & (0.5) & 5 & 8 & 36 \\ (0.5) & (-\frac{1}{6}) & \frac{1}{3} & \frac{1}{3} & -6 \end{bmatrix}$$

and

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

The only multiplier to be calculated is m_{43}

$$m_{43} = \frac{a_{43}}{a_{33}} = \frac{1}{15}$$

and the only row operation to be performed is

$$R4 \rightarrow R4 - m_{43}R3$$

The final matrix is

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & 7 & 9 & -17 \\ (-1) & -3 & -5 & 5 & -21 \\ (0.5) & (0.5) & 5 & 8 & 36 \\ (0.5) & (-\frac{1}{6}) & (\frac{1}{15}) & -0.2 & -8.4 \end{bmatrix}$$

From here, we can extract both the lower and upper triangular matrices \mathbf{L} , \mathbf{U} respectively.

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0.5 & 0.5 & 1 & 0 \\ 0.5 & -\frac{1}{6} & \frac{1}{15} & 1 \end{bmatrix} \mathbf{U} = \begin{bmatrix} 2 & 3 & 7 & 9 \\ 0 & -3 & 5 & 5 \\ 0 & 0 & 5 & 8 \\ 0 & 0 & 0 & -0.2 \end{bmatrix} \mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

- (b) When checked in MATLAB, the results of the hand calculation agree with the computed result.

`lup_check.m` script used to generate matrices.

```
% generate LU decomposition of matrix A

A = [1 2 3 4; 2 3 7 9; -2 -6 -2 -4; 1 0 11 15];
[L,U,P] = lu(A);

%Display results
fprintf('L =\n');
disp(L);

fprintf('\nU =\n');
disp(U);

fprintf('\nP =\n');
disp(P);
```

Output of `lup_check.m`

```
lup_check
L =
    1.0000         0         0         0
   -1.0000    1.0000         0         0
    0.5000    0.5000    1.0000         0
    0.5000   -0.1667    0.0667    1.0000

U =
    2.0000    3.0000    7.0000    9.0000
         0   -3.0000    5.0000    5.0000
         0         0    5.0000    8.0000
         0         0         0   -0.2000

P =
     0     1     0     0
     0     0     1     0
     0     0     0     1
     1     0     0     0
```

(c) If the decomposition is correct, $\mathbf{LU} == \mathbf{AP}$

$$\begin{aligned}
\mathbf{LU} &= \begin{bmatrix} L_{11}U_{11} & L_{11}U_{12} & L_{11}U_{13} & L_{11}U_{14} \\ L_{21}U_{11} & L_{21}U_{12} + L_{22}U_{22} & L_{21}U_{13} + L_{22}U_{23} & L_{21}U_{14} + L_{22}U_{24} \\ L_{31}U_{11} & L_{31}U_{12} + L_{32}U_{22} & L_{31}U_{13} + L_{32}U_{23} + L_{33}U_{33} & L_{31}U_{14} + L_{32}U_{24} + L_{33}U_{34} \\ L_{41}U_{11} & L_{41}U_{12} + L_{42}U_{22} & L_{41}U_{13} + L_{42}U_{23} + L_{43}U_{33} & L_{41}U_{14} + L_{42}U_{24} + L_{43}U_{34} + L_{44}U_{44} \end{bmatrix} \\
&= \begin{bmatrix} 2 & 3 & 7 & 9 \\ -2 & (-3-3) & (-7+5) & (-9+5) \\ 1 & (1.5-1.5) & (3.5+2.5+5) & (4.5+2.5+8) \\ 1 & (1.5+0.5) & (3.5-\frac{5}{6}+\frac{1}{3}) & (4.5-\frac{5}{6}+\frac{8}{15}-0.2) \end{bmatrix} \\
&= \begin{bmatrix} 2 & 3 & 7 & 9 \\ -2 & -6 & -2 & -4 \\ 1 & 0 & 11 & 15 \\ 1 & 2 & 3 & 4 \end{bmatrix}
\end{aligned}$$

The product \mathbf{AP} can be found analytically by taking the row operations needed to convert the identity matrix \mathbf{I} into \mathbf{P} , and applying them to \mathbf{A} .

$$\mathbf{P} = (R1_I \leftrightarrow R2_I) \rightarrow (R2_I \leftrightarrow R3_I) \rightarrow (R3_I \leftrightarrow R4_I)$$

Performing the same operations on \mathbf{A} yields

$$\mathbf{AP} = \begin{bmatrix} 2 & 3 & 7 & 9 \\ -2 & -6 & -2 & -4 \\ 1 & 0 & 11 & 15 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

The two matrices are equal, and there is no residual.